Dynamic Memory Oriented Transformations in the MPEG4 IM1-Player on a Low Power Platform

P. Marchal\textsuperscript{1,2}, C. Wong\textsuperscript{1,2}, A. Prayati\textsuperscript{3}, N. Cossement\textsuperscript{2}, F. Catthoor\textsuperscript{1,2}, R. Lauwereins\textsuperscript{2}, D. Verkest\textsuperscript{1}, and H. De Man\textsuperscript{1,2}

\textsuperscript{1}IMEC, Kapeldreef 75, Leuven, Belgium
tel: 32-16-281202, fax: 32-16-281515, catthoor@imec
\textsuperscript{2}K.U.Leuven-ESAT, Leuven, Belgium
\textsuperscript{3}Department of Electrical and Computer Engineering, University of Patras, Greece

Abstract. In this paper we propose several system-level transformations that allow to reduce the dynamic memory requirements of complex real-time multi-media systems. We demonstrate these transformations on the protocol layer of the MPEG4 IM1-player. As a consequence, up to 20\% of the global power consumption of the protocol subsystem can be eliminated, which is significant due to the programmable processor target. The entire MPEG4 description is assumed to be mapped on a heterogeneous platform combining several software processors and hardware accelerators.

1 Motivation and Context

Rapid evolution in sub-micron process technology allows ever more complex systems to be integrated on one single chip. However, design technologies fall behind these advances in processing technology. A consistent system design technology that can cope with such complexity and with the ever shortening time-to-market requirements is in great need. It should allow to map these applications cost-efficiently to the target architecture while meeting all real-time and other constraints. The design of these complex systems starts with the specification of all its requirements. This is mostly done by a very heterogeneous group of people originating from different communities. They often formalize the specifications into an international standard. Normally, they simultaneously build an executable model of the specified system without investing too much effort in the efficiency of the implementation. Afterwards, this executable model is then translated by another group of designers into a working system. They manually try to improve the initial implementation. As a result of the large complexity and size of these systems, this mostly happens in a very \textit{ad hoc} and time-consuming way. However, rapidly changing features and standards enforce the design trajectory to be shorter. Hence, less time is left to improve the initial specification, resulting in suboptimal, power hungry and not very retargetable systems.
As a consequence, one can conclude that there is a need for a formalized methodology that allows us to systematically improve the initial task-level specification. Therefore, we propose a methodology that starts from an initial, unified specification model that is effectively capable of representing the right system-level abstraction \cite{16,7}. We then systematically transform this model in order to reduce the cost of dynamically allocated data and to increase the amount of concurrency available in the application. We mainly target the protocol layer of multi-media and network applications which are typically mapped to power-hungry programmable processors. The high throughput but more regular and mostly fixed data processing kernels (e.g. IDCT in video-decoders) are assumed to be mapped to accelerators based on more custom processors. This model can then finally be used to map the improved specification on a ‘platform’. In this paper we will focus on the code transformation stage oriented to the memory subsystem. That subsystem contributes heavily to the overall power and area cost. The effect of these transformations on the dynamic memory requirements of the system will be demonstrated, when mapping it on a heterogeneous low power platform that combines parallel software processors and hardware accelerators. By reducing the amount of dynamically allocated memory needed by the applications with our transformations, either the quality of service of the system can be improved or the power consumption and the area (size) of the system can be reduced.

2 Target Application

The target applications of our task-level system synthesis approach are advanced real-time multi-media systems. These applications involve a combination of complex data- and control-flow where complex data types are manipulated and transferred. Most of these applications are implemented on portable devices, putting stringent constraints on the degree of integration and on their power consumption. Secondly, these systems are extremely heterogeneous in nature and combine high performance data processing (e.g. data processing on transmission data input) as well as slow rate control processing (e.g. system control functions), synchronous as well as asynchronous parts, analog and digital, and so on. Thirdly, time-to-market has become a critical factor in the design phase. Finally, these systems are subjected to stringent real-time constraints (both hard and soft deadlines are present), complicating their implementation considerably.

The main driver for our research is the IM1-player \cite{8}. This player is based on the MPEG4 standard, which specifies a system for the communication of interactive audio-visual scenes composed of objects. The player can be partitioned into four layers (see Fig.1): the delivery layer, the synchronization layer, the compression layer and the composition layer. The delivery layer is a generic mechanism that conveys streaming data to the player. Input streams flow through the different layers and are gradually interpreted. We mainly focus on the compression layer. This layer contains several decoders. Two important ones are the BIFS\footnote{BInary Format for Streams}.